REMARKS

Claims remaining in the present application are Claims 1-16 and 27-37. Claims 1-6, 8, 10, 28, 30, and 31 have been amended. No new matter has been added as a result of these amendments.

CLAIM REJECTIONS

35 U.S.C. 102(b)

The rejection states that Claims 1, 3-9, 27-28, 30-32, 34, and 37 are rejected under 35 U.S.C. 102(b) as being anticipated by Rostoker, U.S. Patent No. 5,544067. The rejection is respectfully traversed, for the reasons below. It is respectfully submitted that Claims 1, 3-9, 27-28, 30-32, 34, and 37 are not anticipated or rendered obvious by Rostoker, for the reasons below.

Currently amended Claim 1 recites:

> A method of facilitating a circuit design to be implemented in a programmable device, said method comprising:
>
> a) causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display performed in response to said module being selected;
>
> b) a computer program determining a valid position for said module in a graphical user interface, said graphical user interface having a plurality of resource icons representing said programmable resources, said valid position based on characteristics of said module and characteristics of said programmable resources, said determination made in response to a user request for said valid position for said module in said graphical user interface; and
>
> c) providing in said resource icons an indication of said valid position of said module, said computer program generating said indication.

Claim 1 recites "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program."

Rostoker, as understood by Applicants, may suggest a program for circuit design. Rostoker, as further understood by Applicants, may suggest a software program having a logic verifier to check schematics for design errors. Rostoker may also suggest a logic verifier generate error indications in response to design problems (Column 9, lines 26-30).

However, Rostoker as understood by Applicant does not teach the claimed limitation of "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program."

Rostoker, as understood by Applicant, may suggest (Column 12, lines 30-35) the circuit being designed be frequently simulated and that errors in small portions of the circuit are more readily detected. As further understood by Applicants, Rostoker also suggests during entire circuit simulations, compound errors may occur, masking other errors.

Rostoker may suggest a portion of the software program generating a net list as a summary of connections between the components. Rostoker may

also suggest a logic compiler utilizing the net list, a component database, and layout, verification, and simulation information to generate a schematic object file or files. Rostoker may further suggest a logic verifier checking the schematic for design errors (Column 9, lines 18-28).

Further, if errors are detected by the logic verifier, a designer may have to return to previous steps in the process to correct those design errors For example, when errors are detected during verification of behavioral function, previous steps in the design process that a designer may have to review and correct may include design specification or description (Column 15, lines 46-65), module description (Column 15, lines 66-67 to Column 16, lines 1-5). When errors are detected during verification of structural function, previous steps in the design process that a designer may have to review and correct may include module description, synthesis, and structural description (Column 16, lines 31-51). It is noted that if partitioning (Column 15, lines 53-65) and composition (Column 16, lines 6-10) have not been performed prior to the verification of the structural function, a designer may need to check the design steps as discussed regarding verification of the behavioral function, as designer may need to modify the design description (Column 16, lines 58-61). Thus, Rostoker, as understood by Applicants, does not teach a software program "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program."

Thus, by "providing in said resource icons an indication of said valid position of said module, said indication generated by said computer program," as claimed, embodiments of the instant specification provide an advantage over Rostoker by reducing the instances of design error.

Thus, Applicants respectfully assert that Rosoker does not teach the claimed limitations of element c) of Claim 1.

Claim 1 also recites "said valid position based on characteristics of said module and characteristics of said programmable resources, said determination made in response to a user request for said valid position for said module in said graphical user interface."

> Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim (Lindemann Maschinefabrik GmbH v. American Hoist & Derrick Co., 221 USPQ 481, 485 (Fed. Cir. 1984)).

Applicants respectfully assert that the rejection fails to discuss the claim limitation, "said valid position based on characteristics of said module and characteristics of said programmable resources." Since anticipation requires the presence in a single prior art reference of each and every element of the claim invention, the Applicants respectfully

assert that the rejection fails to present a prima facie case for anticipation. Therefore, the Applicants respectfully assert that the rejection under 35 U.S.C. §102 has been overcome.

Claim 1 further recites "causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display in response to said module being selected."

The office action asserts that Rostoker displays information related to macros and modules (Column 6, lines 62-65; Column 7, lines 35-39). Applicants have read the cited passages and do not understand Rostoker to teach "causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display in response to said module being selected," as claimed.

Rostoker may show in Figures 13-15, a GUI in which are displayed various design alternatives with reference to a system created data-flow representing a circuit design (Column 26, lines 4-10; Figure 13, lower right). As also understood by Applicants, Rostoker displays a designed circuit as interconnected circle and square primitives (Column 26; lines 11-12; Figure 13, upper left; Figure 14, upper middle; Figure 15, lower right). Further,

Rostoker may show a bar graph representing resources consumed in a circuit (Figure 15). Further, as understood by Applicants, Rostoker suggests a plurality of data flow represented circuit designs from which may be displayed an estimated area consumed in and by a design and also suggests a display of functional units that would be consumed in and by the design in the form of bar graph (Column 26, lines 8-17; Figure 15, upper right). Applicants note that upon selection of an alternative design, the display of functional units and area would change to reflect the selection of the alternative design. Thus, as understood by Applicant, Rostoker may show displaying information regarding circuit resources related to a circuit design.

However, Rostoker does not teach the claimed limitation of "causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display in response to said module being selected."

As understood by Applicant, Rostoker does not suggest a manner in which a module from the alternative circuit designs may be selected, nor does Rostoker describe displaying information related to a selected module from an alternative design. Also understood by Applicant, Rostoker does not describe or suggest a manner in which any of the modules represented by circles and/or squares in the primitives are selectable from the GUI, nor does

Rostoker describe displaying information related to a module represented by the circles and squares in the primitives (Figure 13-15). As further understood by Applicant, Rostoker does not suggest a manner in which a module from the circuit represented by the bar graph (Figure 15) can be selected, nor does Rostoker describe displaying information related to a module represented in the bar graph.

As understood by Applicant, Rostoker does not suggest or describe a module that can be selected from within the displays shown in Figures 13-15. Importantly, by virtue of Rostoker not describing the claimed limitation of a "module being selected," Rostoker does not teach "causing to be displayed information related to a module of a plurality of available modules, said module representing a function implementable in programmable resources available in the programmable device, said display in response to said module being selected." Thus, Rostoker, as understood by Applicant, does not teach the limitations of element a) of Claim 1, as recited.

Claim 1 additionally recites "a computer program determining a valid position for said module in a graphical user interface, said graphical user interface having a plurality of resource icons representing said programmable resources."

Rostoker, as understood by Applicants, may show displaying in a GUI alternative circuit designs in which the alternative circuit designs may show estimated area and consumed resources related to the circuit are displayed. Rostoker may also show circle and square primitives representing a circuit design. Rostoker may also show selectable circuit designs.

However, as understood by Applicant, Rostoker does not teach the claimed limitation of "a computer program determining a valid position for said module in a graphical user interface, said graphical user interface having a plurality of resource icons representing said programmable resources, said valid position based on characteristics of said module and characteristics of said programmable resources, said determination made in response to a user request for said valid position for said module in said graphical user interface."

As understood by Applicant, Rostoker does not describe a displayed circuit design from which a module in the circuit design may be selected. As further understood by Applicant, Rostoker does not describe a module that can be selected from within a circuit design displayed in a GUI. More importantly, Rostoker, as understood by Applicant, does not describe a manner in which a valid position of a module can be determined in a circuit design displayed in a GUI in response to a user request.

Thus, as understood by Applicant, Rostoker does not teach the claimed limitation of element b) of Claim 1.

Therefore, as understood by Applicants, Applicants' Claim 1 is not anticipated by Rostoker, nor does Rostoker teach, suggest, or describe the claimed limitations of Claim 1.

For the foregoing rationale, it is respectfully submitted that Claim 1 is not anticipated by Rostoker. As such, allowance of Claim 1 is earnestly solicited.

Claims 3-9 and 27 depend from Claim 1, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 3-9 are not anticipated nor rendered obvious by Rostoker. Allowance of Claims 3-9 and 27 is earnestly solicited.

Claim 28

Claim 28 has been modified to clarify that resource icons represent a layout of resources of the microcontroller.

Currently amended Claim 28 recites:

> A method of facilitating programming a microcontroller, said method comprising:

a) displaying a graphical user interface (GUI) comprising resource icons representing a layout of resources of said microcontroller;

b) displaying information related to a module of a plurality of modules, said module representing a function implementable in said resources, said displaying information performed in response to said module being selected by a user;

c) a computer program determining a valid position for said module in said layout in the GUI reflecting a valid position for said function in said resources of said microcontroller; and

d) displaying said valid position for said module in said layout of the GUI, wherein programming said microcontroller is facilitated.

Rostoker may show a representation of a specific circuit that is currently being designed (see e.g., Figs. 13-15). However this teaching does not show the claim limitation of, "displaying a graphical user interface (GUI) comprising resource icons representing a layout of resources of said microcontroller." Applicants understand the representation that Rostoker displays to be entirely based on the design that was selected. If a user were to select a different design, a new representation would be shown. However, regardless of which circuit is selected, Rostoker does not teach or suggest displaying the claimed, "displaying a graphical user interface (GUI) comprising resource icons representing a layout of resources of said microcontroller."

Rostoker may also show a bar graph in the upper right of Figure 15. However, a bar graph is not the claimed, "displaying a graphical user interface (GUI) comprising resource icons representing a layout of resources of said microcontroller."

Applicants further assert that Rostoker fails to teach or suggest the claimed limitations of, "a computer program determining a valid position for said module in said layout in the GUI reflecting a valid position for said function in said resources of said microcontroller," and "displaying said valid position for said module in said layout of the GUI, wherein programming said microcontroller is facilitated."

For the foregoing reasons, Rostoker fails to teach or suggest the limitations of Claim 28. Therefore, Applicants respectfully request allowance of Claim 28.

Claims 30-37 depend from Claim 28, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 30-37 are not anticipated by Rostoker. As such, allowance of Claims 30-37 is respectfully solicited.

35 U.S.C. 103(a)

Claims 2, 10-12, 14, and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rostoker in view of Comeau et al. (Pub. No. U.S. 2002/0099863) (hereinafter, Comeau). The rejection is respectfully traversed for the reasons below. It is respectfully submitted that Claims 2,

10-12, 14, and 29 are not rendered obvious by Rostoker in view of Comeau, for the reasons presented below.

Claim 2

Currently amended Claim 2, in part, recites:

> d) said computer program generating at least two elements selected from the group consisting of: an application programming interface (API) for programming an operation of said module, source code for realizing said module in said resources, an interrupt vector table having a call to an interrupt service routine for said module, and a data sheet for a circuit comprising selected modules as positioned in said graphical user interface.

Thus, Claim 2 requires a computer program generating at least two elements from the group consisting of: an application programming interface for programming an operation of a module, source code for realizing the module in the resources, an interrupt vector table having a call to an interrupt service routine for the module, and a data sheet for a circuit comprising selected modules as positioned in said graphical user interface (emphasis added).

Applicants respectfully assert that Rostoker and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of "generating an application programming interface (API) for programming an operation of a first of said selected modules." On page 5 of the Office Action, the rejection concedes that Rostoker fails to disclose this limitation. Moreover, Applicants

respectfully submit that Rostoker fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

Comeau, as understood by Applicants, describes in Paragraph [0046] a peripheral application interface 304 as a structure within support layer 202 to provide application developers a consistent view and communicate with peripherals and processors.

However, Comeau, as understood by Applicant, does not alone or when in combination with Rostoker, suggest generating an application programming interface for programming an operation of a user selected module, as recited in Claim 2.

Next, Applicants respectfully assert that Rostoker and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules." On page 5 of the Office Action, the rejection concedes that Rostoker fails to disclose this limitation. Moreover, Applicants respectfully submit that Rostoker fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

Comeau, as understood by Applicant, describes at Paragraph [0035] a plurality of interrupt vector tables (314, 316) in which processor interrupt vector table 316 points to software support interrupt vector table 314 when a virtual machine 120 and a support layer 202 are compiled together with a bootloader 318. Further understood by Applicants, Comeau, at Paragraph [0036], describes software support interrupt vector table 314 and processor interrupt vector table 316 in which 314 overlaps 316. Comeau, as understood by Applicants and as stated in the Office Action, describes at Paragraph [0049] a scheduler for controlling threads to an interrupt service routine. However, Comeau, as understood by Applicants, does not teach or suggest either generating an interrupt vector table or generating an interrupt vector table having a call to an interrupt service routine, as claimed in Claim 2.

Thus, as understood by Applicants, Rostoker, in view of Comeau, does not, alone or in combination, suggest generating an interrupt vector table, nor does Rostoker, in view of Comeau, alone or in combination, suggest generating an interrupt vector table having a call to an interrupt service routine, as claimed in Claim 2.

Rostoker, as understood by Applicants, describes data sheets for any library element (Column 6, lines 66-67 to Column 7, lines 1-3; Column 29, lines 30-31) and data sheets to extract the timing description of a timing

block (Column 18, lines 23-25). However, Rostoker, as understood by Applicant, does not suggest, teach or describe generating a data sheet for a circuit comprising selected modules as positioned in said graphical user interface, as claimed.

Comeau, as understood by Applicants, does not suggest, teach or describe data sheets, nor does Comeau describe generating a data sheet for a circuit comprising selected modules as positioned in said GUI, as clamed. Thus, Comeau does not remedy the shortcomings of Rostoker.

For the above reasonings, Applicants assert that the teachings of Rostoker in view of Comeau do not remedy the shortcomings of Rostoker, nor does the combination Rostoker in view of Comeau teach or suggest the claimed limitation of Claim 2.

Applicants further traverse the rejection on the grounds that one of ordinary skill in the arts would not have been motivated to combine the teachings of Rostoker with the teachings of Comeau.

Rostoker describes, as stated by Examiner, a method and system for creating, deriving, and validating circuit design. As understood by Applicants, Rostoker describes utilizing a high level language, e.g., VHDL (Very ((High Speed Integrated Circuit)) Hardware Description Language),

which implements multiple compilers, e.g., a schematic compiler, a design compiler, a VHDL compiler, and a logic compiler, to enable performing of many of the functionalities contained therein. Thus, as understood by Applicants, Rostoker suggests an environment for compiled language applications.

Comeau, as understood by Applicants, describes a software support layer for processors executing interpreted languages (Title) with specific mention toward application programmers (Abstract). Comeau does not teach, suggest, or describe a software support layer implemented in a circuit designing environment. Further, it is well known in the art that a compiled programming language, such as that suggested by Rostoker, is well suited for high volume and intense computational functions, e.g., circuit design and implementation. It is also well known that an interpreted programming language, e.g., Java, will run slower than a compiled programming language. Thus, Applicants assert that combining the teachings of Rostoker with the teachings of Comeau will have a detrimental effect to the functions desired by Rostoker. Thus, Applicants respectfully traverse Examiners motivation to combine the teachings of Rostoker with the teachings of Comeau.

Claim 10

Currently amended Claim 10 recites:

> A computer readable medium having stored thereon program instructions for implementing a method for assisting circuit designing, said method comprising:
>
> a) determining valid positions in a graphical user interface for user selected modules to be placed in said graphical user interface, said graphical user interface describing resources in which said selected modules are programmably operable, said valid positions based on characteristics of said user selected modules and characteristics of said resources; and
>
> b) generating at least two elements selected from the group consisting of: an application programming interface (API) for programming an operation of a first of said user selected modules, source code for realizing said user selected modules in said resources, an interrupt vector table having a call to an interrupt service routine for a first of said user selected modules, and a data sheet for a circuit comprising said user selected modules as positioned in said graphical user interface.

Claim 10 has been amended to clarify that the module is implementable in programmable resources.

Claim 10 recites, "determining valid positions in a graphical user interface for selected modules to be placed in said graphical user interface, said graphical user interface describing resources operable and in which are implementable said selected modules." For reasons discussed in the response to Claim 1, Rostoker fails to teach or suggest this claimed limitation. Comeau fails to remedy this deficiency. Therefore, neither Rostoker nor Comeau, alone or in combination, teach or suggest this

claimed limitation. As such, Claim 10 is not rendered obvious over the cited combination.

Claim 10 further recites that at least two of the following elements are generated: 1) an application programming interface (API) for programming an operation of a first of said selected modules; 2) source code for realizing said selected modules in said resources; 3) an interrupt vector table having a call to an interrupt service routine for a first of said selected modules; and 4) a data sheet for a circuit comprising said selected modules as positioned in said graphical user interface.

Neither Rostoker nor Comeau teach or suggest, alone or in combination, generating an API application programming interface (API) for programming an operation of a first of said selected modules, an interrupt vector table having a call to an interrupt service routine for a first of said selected modules, or a data sheet for a circuit comprising said selected modules as positioned in said graphical user interface. Therefore, Claim 10 is not rendered obvious by Rostoker in view of Comeau. Continuing with the response to the rejection of Claim 10, the Applicants present the following arguments.

Applicants respectfully assert that Rostoker and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of "generating

an application programming interface (API) for programming an operation of a first of said selected modules." Applicants respectfully submit that Rostoker fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

In support of the rejection, the rejection cites Comeau at paragraph [0043]. On page 5 of the Office Action, the rejection asserts that Comeau recites an apparatus that supports processors executing interpreted language applications that make use of an API. While Comeau may disclose the use of an API, Comeau does not teach or suggest the generation of APIs, as claimed.

Thus, Applicants respectfully submit that even if Comeau were to be combined with Rostoker, the combination fails to teach or suggest "generating an application programming interface (API) for programming an operation of a first of said selected modules," as claimed.

Next, Applicants respectfully assert that Rostoker and Comeau, alone or in combination, fail to teach or suggest the claimed limitation of "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules." The rejection concedes that Rostoker fails to disclose this limitation. Moreover, Applicants respectfully

submit that Rostoker fails to teach or suggest this limitation. Comeau also fails to disclose or suggest this limitation.

In support of the rejection to the presently discussed limitation, the Office Action asserts that Comeau makes use of interrupt vector tables and routines. However, the Office Action fails to assert that Comeau generates such items. Applicants note that Comeau discloses an interrupt vector table. However, Comeau fails to teach or suggest "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules," as claimed. Thus, Applicants respectfully submit that even if Rostoker were to be combined with Comeau, the combination fails to teach or suggest, "generating an interrupt vector table having a call to an interrupt service routine for a first of said selected modules," as claimed.

Rostoker, as understood by Applicants, describes data sheets for any library element (Column 6, lines 66-67 to Column 7, lines 1-3; Column 29, lines 30-31) and data sheets to extract the timing description of a timing block (Column 18, lines 23-25). However, Rostoker, as understood by Applicant, does not suggest, teach or describe a data sheet for a circuit comprising selected modules as positioned in said graphical user interface, as claimed.

Comeau, as understood by Applicants, does not suggest, teach or describe data sheets. Thus, Comeau does not remedy the shortcomings of Rostoker.

For the foregoing reason, at best Rostoker and Comeau teach or suggest generating one of the items, e.g., source code, that Claim 10 recites as being generated. Thus, Rostoker and Comeau, alone or in combination, fail to teach or suggest generating at least two elements selected from the group. Hence, Claim 10 is not rendered obvious by Rostoker in view of Comeau. As such, Applicants earnestly request allowance of Claim 10.

Claims 11, 12, and 14 depend from Claim 10, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 11, 12, and 14 are not rendered obvious by Rostoker in view of Comeau. As such, allowance of Claims 11, 12, and 14 is respectfully solicited.

Claim 29

For the reasons discussed in the response to Claim 10, neither Rostoker nor Comeau, alone or in combination, teach or suggest the claimed limitations of Claim 29. Therefore, allowance of Claim 29 is earnestly solicited.

35 U.S.C. 103(a)

Claims 13, 15, 16, 33, 35, 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rostoker in view of Comeau and further in view of Zizzo (U.S. Patent No. 6,578,174). The rejection is respectfully traversed for the reasons below. It is respectfully submitted that Claims 13, 15, 16, 33, 35, 36 are not rendered obvious by Rostoker in view of Comeau further in view of Zizzo, for the reasons presented below.

Applicants respectfully assert that Rostoker in view of Comeau further in view of Zizzo, alone or in combination, does not suggest, describe, or teach the claimed limitation of Claim 13 which, in part, claims "d1) determining said new valid position for said first of said user selected modules in said graphical user interface, based on an Extensible Markup Language (XML) description of said first of said user selected modules and an XML description of said resources."

The rejection concedes that Rostoker in view of Comeau fails to teach the use of XML description of modules, or HTML datasheets of the modules. Applicants respectfully assert that Rostoker in view of Comeau does not suggest, describe or teach the claimed limitation.

In support of the rejection, on page 6 of the Office Action, the rejection asserts that Zizzo cites using XML and HTML (column 7, lines 47-52;

column 9, lines 20-36). However, the rejection fails to assert that Zizzo determines a new position of a module using XML descriptions of the module and the resource in which the module is to be implemented. Applicants note that Zizzo discloses Extensible Markup Language. However, Zizzo fails to teach, describe, or suggest using XML descriptions of modules and resources to determine valid positions of the modules in the resources, as claimed. Thus, Applicants respectfully submit that Rostoker in view of Comeau in further view of Zizzo does not, alone or in combination, suggest, teach, or describe "determining said new valid position for said first of said user selected modules in said graphical user interface, based on an Extensible Markup Language (XML) description of said first of said user selected modules and an XML description of said resources."

Therefore, Claim 13 is not rendered obvious by Rostoker in view of Comeau further in view of Zizzo. As such, Applicants respectfully request allowance of Claim 13. Claims 15 and 16 depend from Claim 10, which is believed to be allowable for the foregoing reasons. As such, it is respectfully submitted that Claims 15 and 16 are not rendered obvious by Rostoker in view of Comeau in further view of Zizzo. Allowance of Claims 15 and 16 is respectfully solicited.

Claims 33, 35, and 36 depend from Claim 28, which is believed to be allowable for the aforementioned reasonings with reference to Claim 28. As

such, it is respectfully submitted that Claims 33, 35, and 36 are not rendered obvious by Rostoker in view of Comeau in further view of Zizzo. Allowance of Claims 33, 35, and 36 is respectfully solicited.

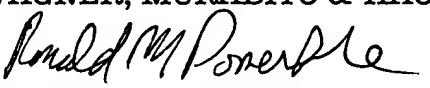
CONCLUSION

In light of the above listed amendments and remarks, reconsideration of the rejected Claims is requested. Based on the arguments and amendments presented above, it is respectfully submitted that Claims 1-16 and 27-37 overcome the rejections of record and, therefore, allowance of Claims 1-16 and 27-37 is earnestly solicited.

Should the Examiner have a question regarding the instant response, the Applicants invite the Examiner to contact the Applicants' undersigned representative at the below listed telephone number.

Respectfully submitted,
WAGNER, MURABITO & HAO LLP

Dated: 6/21, 2004



Ronald M. Pomerenke
Registration No. 43,009

Address: WAGNER, MURABITO, & HAO LLP
Two North Market Street
Third Floor
San Jose, California 95113

Telephone: (408) 938-9060 Voice
(408) 938-9069 Facsimile